Network Working Group
RFC: #735
NIC: #42083

David H. Crocker
Rand-ISD
(Dcrocker at Rand-Unix)
Richard H. Gumpertz
Carnegie-Mellon University
(Gumpertz at CMU-10A)

Obsoletes: RFC #729 (NIC #40306)

3 November 1977

Revised TELNET Byte Macro Option

1. Command name and code:

   BM 19

2. Command Meanings:

   IAC WILL BM

      The sender of this command REQUESTS or AGREES to use the BM option, and will send single data characters which are to be interpreted as if replacement data strings had been sent.

   IAC WON'T BM

      The sender of this option REFUSES to send single data characters which are to be interpreted as if replacement data strings had been sent. Any existing BM <macro byte> definitions are discarded (i.e., reset to their original data interpretations).

IAC DO BM

   The sender REQUESTS or AGREES to have the other side (sender of WILL BM) send single data characters which are to be interpreted as if replacement data strings had been sent.

IAC DON'T BM

   The sender REFUSES to allow the other side to send single data characters which are to be interpreted as if replacement data strings had been sent. Any existing BM <macro byte> definitions are to be discarded.

IAC SB BM <DEFINE> <macro byte> <count>
<replacement string> IAC SE

where:

<macro byte> is the data byte actually to be sent across the network; it may NOT be Telnet IAC (decimal 255, but may be any other 8-bit character.

<count> is one 8-bit byte binary number, indicating how many <replacement string> characters follow, up to the ending IAC SE, but not including it. Note that doubled IACs in the definition should only be counted as one character per pair.

<replacement string> is a string of zero or more Telnet ASCII characters and/or commands, which the <macro byte> is to represent; any character may occur within a <replacement string>. Note, however, that an IAC in the string must be doubled, to be interpreted later as an IAC; to be interpreted later as data byte 255, it must be quadrupled in the original <replacement string> specification.

The indicated <macro byte> will be sent instead of the indicated <replacement string>. The receiver of the <macro byte> (the sender of the DO BM) is to behave EXACTLY as if the <replacement string> string of bytes had instead been received from the network. This interpretation is to occur before any other Telnet interpretations, unless the <macro byte> occurs as part of a Telnet command; in this case no special interpretation is to be made. In particular, an entire Telnet subnegotiation (i.e. from IAC SB through IAC SE) is to be considered a Telnet command in which NO replacement should be done.

The effect of a particular <macro byte> may be negated by reseting it to "expand" into itself.

IAC SB BM <DEFINE> X <0> IAC SE may be used to cause X to be ignored in the data stream.

<DEFINE> is decimal 1.

IAC SB BM <ACCEPT> <macro byte> IAC SE

The receiver of the <DEFINE> for <macro byte> accepts the requested definition and will perform the indicated replacement whenever a <macro byte> is received and is not part of any IAC Telnet command sequence.

<ACCEPT> is decimal 2.

IAC SB BM <REFUSE> <macro byte> <REASON> IAC SE

The receiver of the <DEFINE> for <macro byte> refuses to perform the indicated translation from <macro byte> to <replacement string> because the particular <macro byte> is not an acceptable choice, the length of the <replacement string> exceeds available storage, the length of the actual <replacement string> did not match the length predicted in the <count>, or for some unspecified reason.

<REFUSE> is decimal 3.

<REASON> may be

<BAD-CHOICE>	which is decimal 1;
<TOO-LONG>	(for receiver's storage) which is decimal 2;
<WRONG-LENGTH>	(of actual string compared with promised length in <count>) which is decimal 3; or
<OTHER-REASON>	(intended for use only until this document can be updated to include reasons not anticipated by the authors) which is decimal zero (0).

IAC SB BM <LITERAL> <macro byte> IAC SE

The <macro byte> is to be treated as real data, rather than as representative of the <replacement string>

Note that this subcommand cannot be used during Telnet subcommands, since subcommands are defined to end with the next occurrence of "IAC SE". Including this BM subcommand within any Telnet subcommand would therefore prematurely terminate the containing subcommand.

<LITERAL> is decimal 4.

IAC SB BM <PLEASE CANCEL> <macro byte> <REASON> IAC SE

The RECEIVER of the defined <macro byte> (i.e., the sender of IAC DO BM) requests the sender of <macro byte> to cancel its definition. <REASON> is the same as for the <REFUSE> subcommand.

The <macro byte> sender should (but is not required to) respond by resetting <macro byte> (i.e., sending an IAC SB BM <DEFINE> <macro byte> <1> <macro byte> IAC SE).

If the receiver absolutely insists on cancelling a given macro, the best it can do is to turn off the entire option, with IAC DONT BM, wait for an acknowledging IAC WONT BM and then restart the option, with IAC DO BM. This will reset all other macroes as well but it will allow the receiver to REFUSE with code BAD CHOICE if/when the foreign site attempts to redefine the macro in question.

3. Default:

WON'T BM -- DON'T BM

No reinterpretation of data bytes is done.

4. Motivation for the option:

Subcommands for Telnet options currently require a minimum of five characters to be sent over the network (i.e., IAC SB <Option name> IAC SE). For subcommands which are employed infrequently, in absolute numbers and in relation to normal data, this overhead is tolerable. In other cases, however, it is not. For example, data which is sent in a block- oriented fashion may need a "block separator" mark. If blocks are commonly as small as five or ten bytes, then most of the cross-net data will be control information. The BM option is intended as a simple data compression technique, to remove this overhead from the communication channel.

5. Description of the option

The option is enabled through the standard Telnet Option negotiation process. Afterwards, the SENDER of data (the side which sends the IAC WILL BM) is free to define and use mappings between single and replacement NVT characters. Except for the ability to refuse particular definitions, the receiver of data has no control over the definition and use of mappings.

The sender (of the WILL BM) is prohibited from using or redefining a <macro byte> until it has received an <ACCEPT> <REFUSE>, or DONT BM, in reply to a <DEFINE>.

NOTE: The Telnet command character IAC (decimal 255) may be a member of a <replacement string> but is the ONLY character which may NOT be defined as a <macro byte>.

Within any Telnet command (i.e., any sequence beginning with IAC) macro replacement may NOT take place. Data are to be interpreted only as their normal character values. This avoids the problem of distinguishing between a character which is to be taken as a <macro byte>, and interpreted as its corresponding <replacement string>, and one which is to be taken as its usual Telnet NVT value. In all other cases, however, <macro byte>s are to be interpreted immediately, as if their corresponding <replacement string>s had actually been sent across the network. Expanded strings are not subject to reinterpretation, so that recursive definitions cannot be made. Telnet commands may be included in <replacement strings>; however, they must be totally contained within the macro or must begin within the macro and terminate outside of it. In particular, they may NOT begin outside a macro and continue or terminate inside one, since no macro replacement takes place while processing any Telnet command.

Note that when skipping data due to Telnet SYNCH (INS/DM) processing, BM macro replacement should still take place, since (for example) "IAC DM" would be a valid <replacement string>.

The <count> in the <DEFINE> subcommand is intended to allow the receiver to allocate storage. IAC interpretation is not over-ridden during BM subcommands so that IAC SE will continue to safely terminate malformed subcommands.

The BM option is notably inefficient with regard to problems during <macro byte> definition and use of <macro byte>s as real data. It is expected that relatively few <macro byte>s will be defined and that they will represent relatively short strings. Since the Telnet data space between decimal 128 and decimal 254 is not normally used, except by implementations employing the original (obsolete) Telnet protocol, it is recommended that <macro byte>s normally be drawn from that pool.